**Introduction**
- Machine comprehension is very important for natural language processing (NLP) research [1]
- We propose to replace the regular output layers in the baseline BiDAF with an Answer Pointer [2]
- We also explore and implement QANet [3]
  - Explore both light and complex models
  - Explore both with and without Answer Pointer

**QANet**
QANet is very similar to Transformer but its main component Encoder Block also has stacked convolutional sublayers.

**Discussion**

<table>
<thead>
<tr>
<th>Model</th>
<th>Dev F1</th>
<th>Dev EM</th>
</tr>
</thead>
<tbody>
<tr>
<td>BiDAF Baseline</td>
<td>58.26</td>
<td>55.05</td>
</tr>
<tr>
<td>BiDAF Baseline + Char Embedding</td>
<td>61.52</td>
<td>58.21</td>
</tr>
<tr>
<td>BiDAF Baseline + Char Embedding + Answer Pointer</td>
<td>62.18</td>
<td>59.13</td>
</tr>
<tr>
<td>QANet v1 + Char Embedding</td>
<td>61.89</td>
<td>58.01</td>
</tr>
<tr>
<td>QANet v1 + Char Embedding + Answer Pointer</td>
<td>64.35</td>
<td>60.93</td>
</tr>
<tr>
<td>QANet v2 + Char Embedding + Answer Pointer</td>
<td>68.29</td>
<td>64.98</td>
</tr>
</tbody>
</table>

**Challenges / Limitations**
- Longer answer performs worse than shorter answer
- "Why" question type performs worse than other question types

**Answer Pointer**
The Answer Pointer conditions the probability distribution for the end location on the start location

**Experiment**

**Summary**

- Integrating Answer Pointer to BiDAF and QANet improves F1 and EM performance.
- Larger QANet performance better than lighter QANet.
- Future Work
  - Explore Transformer-XL for longer-term dependencies.
  - Explore models with low memory use like Reformer.

References: